4 means for delaying said data stream portion; and

5 means for accelerating a second data stream portion that is preceded by said first

6 data stream portion.

Please cancel claims 3 without prejudice or disclaimer and insert the following new claims:

1 4. A method for avoiding overflow of a decoder buffer containing a portion of new data

2 stream and a portion of an old data stream, comprising:

3 (a) determining a total amount of old data stream data that, if transmitted to said

4 decoder buffer, would occupy said decoder buffer;

5 (b) adding to said total amount, an amount of new data stream data to obtain a

6 combined amount of data;

7 (c) testing if said combined amount of data would overflow said decoder buffer; and

8 (d) if overflow would occur, then causing a portion of the new data stream to be

9 delayed by a delay amount corresponding to at least said overflow, if said portion were to be

10 transmitted to said decoder buffer.

1 5. A method according to claim 4, wherein said step (a) of determining is preceded by

2 determining a maximum size of said decoder buffer;

1 6. A method according to claim 5, wherein said maximum size is determined according to a

2 buffer size parameter within the old data stream.

1 7. A method according to claim 5, wherein said maximum size of step (a) is determined

2 according to a buffer size parameter within the new data stream.

1 8. A method according to claim 4, further comprising:

2 prior to testing of step (c), subtracting, from said total amount, an amount of old

3 data stream data that, if transmitted, would be decoded by said decoder;

1 9. A method according to claim 4, wherein said delay amount of step (d) is a function of an

2 amount of data stream data by which said buffer is overflowed within said portion of the new

3 data stream.

1 10. A method according to claim 4, wherein said delay amount of step (d) is a function of an

2 amount of data stream data by which said buffer is overflowed in a single instance of

3 overflow within said portion of the new data stream.

1 11. A method according to claim 4, wherein said delay amount of step (d) is a function of a

2 duration of overflow within said portion of the new data stream.

1 12. A method according to claim 4, wherein said delay amount of step (d) is a function of a

2 duration of a single instance of overflow within said portion of the new data stream .

1 13. A method according to claim 4, wherein said delay amount of step (d) is a function of a

2 longest duration instance of overflow within said portion of the new data stream.

1   14. A method according to claim 4, wherein said delay amount of step (d) is equal to a

2   number of data packets of said portion during a longest duration instance of overflow within

3   said portion of the new data stream.

1   15. A method according to claim 4, wherein step (d) further comprises:

2   causing a subsequent portion of said new data stream to be accelerated by an

3   acceleration amount corresponding to said delay amount, if the new data stream is

4   transmitted.

16. A method for detecting overflow of a data stream decoder during splicing of data stream

2   portions including an old data stream portion and a new data stream portion, comprising:

3   (a) determining a plurality of old data stream frame sizes and decoding times

4   corresponding to old data stream frames of said old data stream portion, and storing said

5   frame sizes and said decoding times in a splice-table;

6   (b) determining a maximum decoder buffer size;

7   (c) determining a new frame size and decoding time corresponding to a new data

8   stream frame of the new data stream portion;

9   (d) determining an intermediate size by summing a plurality of old data stream

10   frame sizes stored in the splice table;

11   (e) determining a total size by adding to said intermediate size, the new data stream

12   frame size; and

13   (f) testing for overflow by determining whether said total size exceeds said

14   maximum decoder buffer size.

1   17. A method according to claim 16, wherein said old data stream frame sizes of step (d)

2   include all frames of the old data stream portion that will remain un-decoded when said new

3   data stream frame will be received by the decoder, if the data stream portions are transmitted.

1   18. A method according to claim 17, wherein the data stream portions are transmitted.

1   19. A method according to claim 16, further comprising:

2   (i) if overflow is found in step (f), then causing a transmission time of a portion of

3   new data stream data including said new data stream frame to be delayed.

1   20. A method for correcting overflow of a digitally encoded data stream decoder during

2   splicing of data stream portions including an old data stream portion and a new data stream

3   portion, comprising causing a delay of a scheduled transmission time of a portion of new data

4   stream data.

1   21. A method according to claim 20, wherein said delay is caused by adding null packets to

2   said new data stream portion.

1   22. A method according to claim 20, wherein said delay is caused by re-scheduling

2   transmission of said portion according to a formula:

3

4   (currently scheduled transmission time for said portion) + ((n packets x m

5   bits/packet x multiplexer bit rate) / (data stream bit rate)),

6

7   wherein n indicates a number of packets by which transmission is to be delayed, and m

8   indicates a number of bits in a packet of data stream data to be transmitted.

1   23. A method according to claim 22, wherein m equals 1504.

1   24. A method for splicing digitally encoded data streams, including an old data stream and a

2   new data stream, comprising:

3   (a) modifying a current timing reference of the new data stream to correspond with a

4   splice-out point of the old data stream and a splice-in point of the new data stream, thereby

5   forming a modified new data stream timing reference; and

6   (b) aligning a portion of the new data stream with a portion of the old data stream

7   according to said modified new data stream timing reference, such that a transition from the

8   old data stream to the new data stream, during playback, will be substantially imperceptible.

1   25. A method according to claim 24, wherein said modified new data stream timing reference

2   of step (a) further corresponds with a timing gap between a first decoding time for decoding a

3   last frame of the old data stream and a second decoding time for decoding a first frame of the

4   new data stream.

1   26. A method according to claim 24, wherein determining said modified new data stream

2   timing reference includes:

3   (i) determining said current timing reference of the new data stream;

4   (ii) determining a delay between said current timing reference and a current decoding

5   time of a frame of the new data stream;

6   (iii) determining a new decoding time of said frame of the new data stream that

7   corresponds with a sum of said current decoding time and an inter-frame delay between a

8   decoding time for decoding a last frame of the old data stream and a decoding time for

9   decoding a first frame of the new data stream; and

10   (iv) determining said modified new data stream timing reference as said new decoding

11 time of step (iii) minus said delay of step (ii).

27. A method according to claim 24, wherein determining said modified new data stream

2   timing reference includes:

3   (i) determining a program clock reference of a first packet of said new data stream;

4   (ii) determining a delay between transmission of a first sequence header of said new

5   data stream and a first decode time stamp ("DTS") of a first frame of said new data stream;

6   (iii) determining a continuous DTS as a sum of said first DTS and an inter-frame

7   delay; and

8   (iv) determining a new data stream real-time transmit time as said continuous DTS of

9   step (iii) minus said delay of step (ii).

1   28. A method according to claim 24, wherein said aligning in step (b) sets a start time for

2   transmitting a portion of the new data stream that corresponds with a decoding time for

3   decoding a portion of the old data stream.

1   29. A method according to claim 24, wherein said aligning in step (b) sets a start time for a

2   decoder buffer to begin receiving a portion of the new data stream that corresponds with a

3   decoding time for decoding a portion of the old data stream.

1   30. A method according to claim 24, further comprising:

2   (d) detecting a decoder buffer overflow condition that will result from said splicing, if

3   the data streams are transmitted; and

4   (e) correcting said overflow condition.

1   31. A method according to claim 24, wherein said determining of step (a) is preceded by

2   (i) determining a splice-out point of the old data stream; and

3   (ii) determining a splice-in point of the new data stream.

1   32. A method according to claim 31, wherein step (ii) includes, if an initial frame of the new

2   data stream is of a type that is ordinarily decoded with reference to decoding of a prior frame,

3   then modifying the new data stream to remove said reference.

33. A method according to claim 32, wherein said frame type is selected from a group

2   consisting of B-frames and P-frames, and wherein said step of modifying comprises closing

3   an open group of pictures ("GOP").

1   34. A method according to claim 31, wherein said data streams include video and audio data,

2   wherein step (a) includes determining a video splice-out point and an audio splice-out point,

3   and wherein step (b) includes determining a video splice-in point and an audio splice-in point.

1   35. A method according to claim 31, wherein said splice-out point of step (i) is determined

2   within a user-selectable portion of the old data stream.

1   36. A method according to claim 31, wherein said splice-in point of step (ii) is determined

2   within a user-selectable portion of the new data stream.

1   37. A method according to claim 31, wherein said splice-out point of step (i) is user-

2   selectable.

1   38. A method according to claim 31, wherein said splice-in point of step (ii) is user-

2   selectable.

1   39. A method according to claim 24, wherein step (a) is preceded by determining a first

2   source for the old data stream and a second source for the new data stream.

40. A method according to claim 39, wherein said sources include source types selected from

a group consisting of a storage device, a satellite receiver, a cable receiver, a network, an

audio source, a video source and an encoder.

41. A method according to claim 40, wherein said first source and said second source are of

a same source type.

42. A method according to claim 24, wherein at least one of said data streams is MPEG

encoded.

43. A method according to claim 24, wherein said splicing is accomplished in real-time.

44. A method according to claim 24, wherein step (a) is followed by transmitting a portion of

~~the old data stream.~~

45. A method according to claim 24, wherein step (b) is followed by transmitting a portion of

the new data stream.

46. A data spliced data stream combination comprising an old data stream and a new data

stream spliced together according to the method of claim 24.

47. A computer-readable storage medium storing program code for causing a computer to

perform the steps of:

3 (a) determining a splice-out point within an old data stream;

4 (b) determining a splice-in point within a new data stream; and

5 (c) determining a new data stream real-time transmit start time.

1 48. A computer-readable storage medium according to claim 47, wherein step (a) is preceded

2 by:

3 determining a new data stream pair to be spliced contemporaneously with another data

4 stream pair; and

5 initiating program code for splicing said new data stream pair.

1 49. A computer-readable storage medium according to claim 47, wherein step (a) is preceded

2 by:

3 creating at least one data storage structure for storing portions of said data streams;

4 and

5 storing portions of said data streams in said at least one data storage structure.

1 50. A computer-readable storage medium according to claim 49, wherein said at least one

2 data storage structure is located in memory of a host processing system.

1 51. A method for splicing digitally encoded data streams, including an old data stream and a

2 new data stream, comprising:

3 (a) receiving a user-selectable parameter indicating a portion of the old data stream

4 within which a splice-out point is to be determined;

5   (b) assigning a splice-buffer for storing an old data stream portion and a new data

6   stream portion;

7   (c) directing the old data stream portion to said splice-buffer;

8   (d) determining said splice-out point;

9   (e) directing the new data stream portion to said splice-buffer;

10   (f) determining a splice-in point within the new data stream portion and, if an initial

11   frame of the new data stream portion is dependent upon a frame that precedes the new data

12   stream portion, then modifying the new data stream portion to remove said dependency;

13   (g) if, upon transmission, a decoder buffer would begin to receive the new data stream

14   after said buffer finally receives a portion of the old data stream, then aligning the new data

15   stream with said finally receiving, and

16   (h) if, upon transmission, a decoder buffer would begin to receive the new data stream

17   before said buffer finally receives a portion of the old data stream, then aligning the new data

18   stream with said finally receiving.

1   52. A method according to claim 51, wherein said dependency of step (f) is an open GOP and

2   wherein said modifying closes the open group of pictures ("GOP").

1   53. A method according to claim 51, further comprising:

2   (j) checking for overflow of said decoder buffer; and

3   (k) if overflow is found, then removing said overflow.

1   54. A splicer for splicing digitally encoded data streams, including an old data stream and a

2   new data stream, comprising:

3   (a) means for determining, in accordance with a splice-out point of an old data stream

4   and a splice-in point of a new data stream, a new data stream real-time transmit start time;

5   and

6   (b) means for aligning the new data stream with the old data stream according to said

7   new data stream real-time transmit time.

1   55. A method for preparing a digitally encoded data stream for splicing, comprising:

2   (a) determining a splice-in point of the new data stream; and

3   (b) closing an initial open group of pictures ("GOP") of the new data stream, if the

4   new data stream includes an initial open GOP.

1   56. A splicer for splicing digitally encoded data streams including an old data stream and a

2   new data stream, comprising:

3   (a) means for determining a splice-in point of the new data stream; and

4   (b) means for closing an open group of pictures ("GOP") of the new data stream, if the

5   new data stream includes an open GOP.

1   57. A method according to claim 51, wherein said splice-out point is determined in step (a)

2   according to a user selection between an insert mode option and a splice-only mode option.

1   58. A method according to claim 57, wherein a splice-out point is determined as immediately

2   prior to a sequence header.

1 59. A method according to claim 57, wherein a splice-out point is determined as immediately

2 prior to a first occurring one of a group of pictures ("GOP") header, an I-frame and a P-frame.

1 60. A method according to claim 51, wherein said determining a splice-in point comprises:

2   finding a decode time stamp ("DTS") for a frame of the new data stream, said frame

3 being included within a group of pictures ("GOP") of the new data stream;

4   finding a corresponding presentation time stamp for said frame; and

5   if said frame is other than an I-frame, then closing said GOP.

1 61. A method according to claim 60, wherein said frame is an initial frame of the new data

2 stream.

1 62. A method according to claim 60, wherein finding said DTS includes parsing a first

2 portion of the new data stream for a first sequence header, and then further parsing said first

3 portion for a last DTS before a first frame header.

1 63. A method for assuring that an initial frame of an encoded data stream portion can be

2 independently decoded, comprising:

3   (a) determining an independently decodable frame within said portion;

4   (b) causing playback of said portion to begin with said independently decodable

5 frame; and

6   (c) modifying an ordering parameter of said portion such that a receiving decoder will

7   decode said independently decodable frame as a first frame of said portion.

Subb 16

64. A method according to claim 63, wherein step (b) is accomplished by deleting a frame

2   within said portion that precedes said independently decodable frame.

1   65. A method for closing an open GOP of a digitally encoded data stream, said GOP

2   including a plurality of frames, comprising:

3   (a) determining a first I-frame within said GOP;

4   (b) determining, within said GOP, a largest DTS of all of said frames that precede said

5   I-frame;

6   (c) deleting all frames within said GOP that precede said I-frame;

7   (d) modifying temporal references for at least one remaining frame within said GOP;

8   and

9   (e) replacing a DTS of said I-frame with said largest DTS of step (b).

Subc 11

1   66. A method according to claim 63, wherein step (d) of modifying includes replacing

2   increasing temporal reference values of remaining frames within said GOP with

3   correspondingly increasing temporal reference values of frames deleted in step (c).

1   67. A method for aligning a splice-out portion of a digitally encoded old data stream with a

2   splice-in portion of a digitally encoded new data stream comprising finding a new data stream

3   real-time transmit time.

68. A method according to claim 67 wherein said step of finding includes:

   (a) determining a program clock reference ("PCR") of a first packet of said new data stream;

   (b) determining a delta-period between transmission of a first sequence header of said new data stream and a first decode time stamp ("DTS") of a first frame of said new data stream, if said new data stream is transmitted;

   (c) determining a continuous DTS as a sum of said first DTS and an inter-frame delay; and

   (d) determining said new data stream real-time transmit time as a difference between said continuous DTS and said delta-period.

69. A method according to claim 67 wherein said step of finding is accomplished in real-time during splicing of said new data stream with said old data stream.

70. A method according to claim 68 wherein said inter-frame delay equals 1001/30,000 seconds.

71. A method for aligning a splice-out portion of a digitally encoded old data stream with a splice-in portion of a digitally encoded new data stream, comprising setting a start of receipt time of said new data stream at which, if said new data stream is transmitted, then said new data stream will begin to be received by a decoder in alignment with a decoding time for said splice-out portion of said old data stream.

72. A method according to claim 71 wherein said step of setting includes:

   if, upon transmission of said data streams, said new data stream would begin to be

received by a decoder before the decoder would have received all of said splice-out portion,

then setting a transmission delay parameter for said new data stream.

73. A method according to claim 72 that further includes inserting a number of null packets

corresponding with said delay parameter into said new data stream at a position such that said

null packets will be transmitted substantially prior to other new data stream data, if said new

data stream is transmitted.

74. A method according to claim 73 wherein said number of null packets equals a number of

data packets that, without said inserting, would be received by a decoder before the decoder

has received all of said splice-out portion, if the new data stream is transmitted.

75. A method according to claim 71 wherein said step of setting includes:

   if said new data stream, upon transmission, would begin to be received by a decoder

after the decoder has received all of said splice-out portion, then setting a transmission

acceleration parameter for said new data stream.

76. A method according to claim 75 that further includes deleting a number of null packets

corresponding with said acceleration parameter from a first transmitted portion of said new

data stream, if said new data stream is transmitted.

77. A method according to claim 76 wherein said number of null packets equals a number of data packets that, without said deleting, would be received by a decoder after the decoder has received all of said splice-out portion, if the new data stream is transmitted.

78. A method for aligning a splice-out portion of a digitally encoded old data stream with a splice-in portion of a digitally encoded new data stream, said splice-out portion and said splice-in portion each comprising a plurality of packets, which comprises:

(a) parsing said splice-out portion for a program clock reference ("PCR") of a last packet of said splice-out portion to be transmitted;

(b) parsing said splice-in portion for a first sequence header and a first decode time stamp ("DTS") of a first frame of said new data stream;

(c) determining a continuous DTS of said new data stream;

(d) if the splice-out PCR of step (a) is less than the real-time transmit time of step (c), then storing a value indicating a total number of null packets which, when transmitted prior to said splice-in portion, will cause transmission of said splice-in portion to begin at substantially a same time as decoding of said splice-out portion; and

(e) if said splice-out portion PCR of step (a) is greater than said real-time transmit time of step (c), then storing a total number of null packets which, when deleted from said splice-in portion, will approximate a condition in which the splice-out portion PCR equals the real-time transmit time.

79. A digitally encoded data stream transmitter comprising:

shifting means for determining an amount by which scheduled transmission times of

3   data stream portions are to be accelerated and delayed; and

4   transmitting means for transmitting said data stream portions at transmission times

5   accelerated and delayed by an amount determined by said shifting means.

1   80. An apparatus according to claim 79, wherein said new data stream data is received as a

2   plurality of data packets.

1   81. An apparatus according to claim 80, wherein said amount is calculated as a time

2   corresponding to a number of data packets of data stream data.

1   82. A transmitter for transmitting a digitally encoded new data stream as spliced to a digitally

2   encoded old data stream, comprising:

3   a transmitter; and

4   a bit-clock means for scheduling the transmission of new data stream data of said new

5   data stream at a time corresponding to splicing of said new data stream to said old data

6   stream.